**Exercise 2.5: Docker Networking**

We will use Docker networking, and see how it is configured in a container.

**Step 1**

Connect to the Google Compute Engine virtual machine.

You will need two SSH sessions for this exercise.

cd  
git clone https://github.com/simplilearn-devops/devops-lesson-2

**Step 2**

Change to the exercise directory.  
cd  
cd devops-lesson-2/lab-2.5

Create a Centos Docker container and install net tools.  
docker run -it --name centos centos /bin/bash  
yum install -y net-tools

Check the IP address and hostname.  
ifconfig  
cat /etc/hosts  
hostname

Exit the container using control-D.

Commit the container to an image.  
docker commit centos centos-net  
docker images  
docker rm centos

**Step 3**

Create a bridge network and find its IP range.  
docker network create exnet  
docker network ls  
docker network inspect exnet

Run the centos container with the new network.  
docker run -it --rm --network exnet centos-net /bin/bash

Check the IP address and hostname.  
ifconfig  
cat /etc/hosts  
hostname

Exit the container with control-D.

**Step 4**

Start a new container using the default network.  
docker run -it --rm --name centos centos-net /bin/bash

Check the IP address and hostname.  
ifconfig  
cat /etc/hosts  
hostname

From the second SSH window, connect the network to the container.  
docker network connect exnet centos

Go back to the running container, and see that it now has two IP addresses.  
ifconfig  
cat /etc/hosts  
hostname

Go to the second SSH window and disconnect the network.  
docker network disconnect exnet centos

Go back to the running container, and see that it now has one IP address.  
ifconfig  
cat /etc/hosts  
hostname

Exit the container with control-D.